Express Mail Label No. EV 396082090 US

[0001] REDUCED COMPLEXITY SLIDING WINDOW BASED EQUALIZER

[0002] CROSS REFERENCE TO RELATED APPLICATION(S)

[0003] This application claims priority from U.S. provisional application no. 60/452,165, filed on March 3, 2003, which is incorporated by reference as if fully set forth.

[0004] FIELD OF INVENTION

[0005] The invention generally relates to wireless communication systems, In particular, the invention relates to data detection in such systems.

[0006] BACKGROUND

[0007] Due to the increased demands for improved receiver performance, many advanced receivers use zero forcing (ZF) block linear equalizers and minimum mean square error (MMSE) equalizers.

[0008] In both these approaches, the received signal is typically modeled per Equation 1.

$$\mathbf{r} = \mathbf{H}\mathbf{d} + \mathbf{n} \qquad \text{Equation 1}$$

[0009] $\mathbf{r}$ is the received vector, comprising samples of the received signal. $\mathbf{H}$ is the channel response matrix. $\mathbf{d}$ is the data vector. In spread spectrum systems, such as code division multiple access (CDMA) systems, $\mathbf{d}$ is the spread data vector. In CDMA systems, data for each individual code is produced by despreading the estimated data vector $\mathbf{d}$ with that code. $\mathbf{n}$ is the noise vector.

[0010] In a ZF block linear equalizer, the data vector is estimated, such as per Equation 2.

$$\mathbf{d} = (\mathbf{H})^{-1}\mathbf{r}$$

Equation 2

[0011] $(\cdot)^H$ is the complex conjugate transpose (or Hermetian) operation. In a MMSE block linear equalizer, the data vector is estimated, such as per Equation 3.

$$\mathbf{d} = \left(\mathbf{H}^H\mathbf{H} + \sigma^2\mathbf{I}\right)^{-1}\mathbf{r}$$

Equation 3

[0012] In wireless channels experiencing multipath propagation, to accurately detect the data using these approaches requires that an infinite number of received samples be used. One approach to reduce the complexity is a sliding window approach. In the sliding window approach, a predetermined window of received samples and channel responses are used in the data detection. After the initial detection, the window is slid down to a next window of samples. This process continues until the communication ceases.

[0013] By not using an infinite number of samples, an error is introduced into the data detection. The error is most prominent at the beginning and end of the window, where the effectively truncated portions of the infinite sequence have the largest impact. One approach to reduce these errors is to use a large window size and truncate the results at the beginning and the end of the window. The truncated portions of the window are determined in previous and subsequent windows. This approach has considerable complexity. The large window size leads to large dimensions on the matrices and vectors used in the data estimation. Additionally, this approach is not computationally efficient by detection data at the beginning and at the ends of the window and then discarding that data.

[0014] Accordingly, it is desirable to have alternate approaches to data detection.

[0015] SUMMARY

[0016] Data estimation is performed in a wireless communications system. A received vector is produced. For use in estimating a desired portion of data of the

received vector, a past, a center and a future portion of a channel estimate matrix is determined. The past portion is associated with a portion of the received signal prior to the desired portion of the data. The future portion is associated with a portion of the received vector after the desired portion of the data and the center portion is associated with a portion of the received vector associated with the desired data portion. The desired portion of the data is estimated without effectively truncating detected data. The estimating the desired portion of the data uses a minimum mean square error algorithm having inputs of the center portion of the channel estimate matrix and a portion of the received vector. The past and future portions of the channel estimate matrix are used to adjust factors in the minimum mean square error algorithm.

[0017] BRIEF DESCRIPTION OF THE DRAWING(S)

[0018] Figure 1 is an illustration of a banded channel response matrix.

[0019] Figure 2 is an illustration of a center portion of the banded channel response matrix.

[0020] Figure 3 is an illustration of a data vector window with one possible partitioning.

[0021] Figure 4 is an illustration of a partitioned signal model.

[0022] Figure 5 is a flow diagram of sliding window data detection using a past correction factor.

[0023] Figure 6 is a receiver using sliding window data detection using a past correction factor.

[0024] Figure 7 is a flow diagram of sliding window data detection using a noise auto-correlation correction factor.

[0025] Figure 8 is a receiver using sliding window data detection using a noise auto-correlation correction factor.

[0026] DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT(S)

[0027] Hereafter, a wireless transmit/receive unit (WTRU) includes but is not limited to a user equipment, mobile station, fixed or mobile subscriber unit, pager, or any other type of device capable of operating in a wireless environment. When referred to hereafter, a base station includes but is not limited to a Node-B, site controller, access point or any other type of interfacing device in a wireless environment.

[0028] Although reduced complexity sliding window equalizer is described in conjunction with a preferred wireless code division multiple access communication system, such as CDMA2000 and universal mobile terrestrial system (UMTS) frequency division duplex (FDD), time division duplex (TDD) modes and time division synchronous CDMA (TD-SCDMA), it can be applied to various communication system and, in particular, various wireless communication systems. In a wireless communication system, it can be applied to transmissions received by a WTRU from a base station, received by a base station from one or multiple WTRUs or received by one WTRU from another WTRU, such as in an ad hoc mode of operation.

[0029] The following describes the implementation of a reduced complexity sliding window based equalizer using a preferred MMSE algorithm. However, other algorithms can be used, such as a zero forcing algorithm. $h(\cdot)$ is the impulse response of a channel. $d(k)$ is the k^{th} transmitted sample that is generated by spreading a symbol using a spreading code. It can also be sum of the chips that are generated by spreading a set of symbols using a set of codes , such as orthogonal codes. $r(\cdot)$ is the received signal. The model of the system can expressed as per Equation 4.

$$r(t) = \sum_{k=-\infty}^{\infty} d(k)h(t-kT_c) + n(t) \qquad -\infty < t < \infty$$

Equation 4

[0030] $n(t)$ is the sum of additive noise and interference (intra-cell and inter-cell). For simplicity, the following is described assuming chip rate sampling is used at the

receiver, although other sampling rates may be used, such as a multiple of the chip rate. The sampled received signal can be expressed as per Equation 5.

$$r(j)=\sum_{k=-\infty}^{\infty} d(k)h(j-k)+n(j)=\sum_{k=-\infty}^{\infty} d(j-k)h(k)+n(j) \qquad j\in\{\ldots,-2,-1,0,1,2,\ldots\}$$

Equation 5

T_c is being dropped for simplicity in the notations.

[0031] Assuming $h(\cdot)$ has a finite support and is time invariant. This means that in the discrete-time domain, index L exists such that $h(i)=0$ for $i<0$ and $i\geq L$. As a result, Equation 5 can be re-written as Equation 6.

$$r(j)=\sum_{k=0}^{L-1} h(k)d(j-k)+n(j) \qquad j\in\{\ldots,-2,-1,0,1,2,\ldots\}$$

Equation 6

[0032] Considering that the received signal has M received signals $r(0),\cdots,r(M-1)$, Equation 7 results.

$$\mathbf{r}=\mathbf{Hd}+\mathbf{n}$$

where,

$$\mathbf{r}=[r(0),\cdots,r(M-1)]^T\in C^M,$$

$$\mathbf{d}=[d(-L+1),d(-L+2),\ldots,d(0),d(1),\ldots,d(M-1)]^T\in C^{M+L=1}$$

$$\mathbf{n}=[n(0),\cdots,n(M-1)]^T\in C^M$$

$$\mathbf{H}=\begin{bmatrix} h(L-1) & h(L-2) & \cdots & h(1) & h(0) & 0 & \cdots & \cdots \\ 0 & h(L-1) & h(L-2) & \cdots & h(1) & h(0) & 0 & \cdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \cdots & \cdots & 0 & h(L-1) & h(L-2) & \cdots & h(1) & h(0) \end{bmatrix}\in C^{M\times(M+L-1)}$$

Equation 7

[0033] Part of the vector $\mathbf{d}$ can be determined using an approximate equation. Assuming $M>L$ and defining $N=M-L+1$, vector $\mathbf{d}$ is per Equation 8.

$$\mathbf{d}=[\underbrace{d(-L+1),d(-L+2),\ldots,d(-1)}_{L-1},\underbrace{d(0),d(1),\ldots,d(N-1)}_{N},\underbrace{d(N),\ldots,d(N+L-2)}_{L-1}]^T\in C^{N+2L-2}$$

Equation 8

[0034] The **H** matrix in Equation 7 is a banded matrix, which can be represented as the diagram in Figure 1. In Figure 1, each row in the shaded area represents the vector $[h(L-1), h(L-2), ..., h(1), h(0)]$, as shown in Equation 7.

[0035] Instead of estimating all of the elements in **d**, only the middle N elements of **d** are estimated. $\tilde{\mathbf{d}}$ is the middle N elements as per Equation 9.

$$\tilde{\mathbf{d}} = [d(0), ..., d(N-1)]^T$$

Equation 9

[0036] Using the same observation for **r**, an approximate linear relation between **r** and $\tilde{\mathbf{d}}$ is per Equation 10.

$$\mathbf{r} = \tilde{\mathbf{H}}\tilde{\mathbf{d}} + \mathbf{n}$$

Equation 10

[0037] Matrix $\tilde{\mathbf{H}}$ can be represented as the diagram in Figure 2 or as per Equation 11.

$$\tilde{\mathbf{H}} = \begin{bmatrix} h(0) & 0 & \cdots & \\ h(1) & h(0) & \ddots & \\ \vdots & h(1) & \ddots & 0 \\ h(L-1) & \vdots & \ddots & h(0) \\ 0 & h(L-1) & \ddots & h(1) \\ \vdots & 0 & \ddots & \vdots \\ & \vdots & \ddots & h(L-1) \end{bmatrix}$$

Equation 11

[0038] As shown, the first L-1 and the last L-1 elements of **r** are not equal to the right hand side of the Equation 10. As a result, the elements at the two ends of vector $\tilde{\mathbf{d}}$ will be estimated less accurately than those near the center. Due to this property, a sliding window approach is preferably used for estimation of transmitted samples, such as chips.

[0039] In each, k^{th} step of the sliding window approach, a certain number of the received samples are kept in **r** $[k]$ with dimension N+L-1. They are used to estimate a

set of transmitted data $\tilde{\mathbf{d}}[k]$ with dimension N using equation 10. After vector $\tilde{\mathbf{d}}[k]$ is estimated, only the "middle" part of the estimated vector $\hat{\tilde{\mathbf{d}}}[\mathbf{k}]$ is used for the further data processing, such as by despreading. The "lower" part (or the later in-time part) of $\tilde{\mathbf{d}}[k]$ is estimated again in the next step of the sliding window process in which $\mathbf{r}$ $[k+1]$ has some of the elements $\mathbf{r}$ $[k]$ and some new received samples, i.e. it is a shift (slide) version of $\mathbf{r}$ $[k]$.

[0040] Although, preferably, the window size N and the sliding step size are design parameters, (based on delay spread of the channel (L), the accuracy requirement for the data estimation and the complexity limitation for implementation), the following using the window size of Equation 12 for illustrative purposes.

$$N = 4N_s \times SF$$

Equation 12

SF is the spreading factor. Typical window sizes are 5 to 20 times larger than the channel impulse response, although other sizes may be used.

[0041] The sliding step size based on the window size of Equation 12 is, preferably, $2N_s \times SF$. $N_s \in \{1,2,...\}$ is, preferably, left as a design parameter. In addition, in each sliding step, the estimated chips that are sent to the despreader are $2N_s \times SF$ elements in the middle of the estimated $\hat{\mathbf{d}}[k]$. This procedure is illustrated in Figure 3.

[0042] One algorithm of data detection uses an MMSE algorithm with model error correction uses a sliding window based approach and the system model of Equation 10.

[0043] Due to the approximation, the estimation of the data, such as chips, has error, especially, at the two ends of the data vector in each sliding step (the beginning and end). To correct this error, the $\mathbf{H}$ matrix in Equation 7 is partitioned into a block row matrix, as per Equation 13, (step 50).

$$\mathbf{H} = \left[\mathbf{H}_p \mid \tilde{\mathbf{H}} \mid \mathbf{H}_f\right]$$

Equation 13

[0044] Subscript "p" stands for "past", and "f" stands for "future". $\tilde{\mathbf{H}}$ is as per Equation 10. $\mathbf{H}_p$ is per Equation 14.

$$\mathbf{H}_p = \begin{bmatrix} h(L-1) & h(L-2) & \cdots & h(1) \\ 0 & h(L-1) & \cdots & h(2) \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & h(L-1) \\ 0 & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & 0 \end{bmatrix} \in C^{(N+L-1)\times(L-1)}$$

Equation 14

[0045] $\mathbf{H}_f$ is per Equation 15.

$$\mathbf{H}_f = \begin{bmatrix} 0 & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \cdots & 0 \\ h(0) & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & 0 \\ h(L-3) & \cdots & h(0) & 0 \\ h(L-2) & h(L-3) & \cdots & h(0) \end{bmatrix} \in C^{(N+L-1)\times(L-1)}$$

Equation 15

[0046] The vector $\mathbf{d}$ is also partitioned into blocks as per Equation 16.

$$\mathbf{d} = \left[\mathbf{d}_p^T \mid \tilde{\mathbf{d}}^T \mid \mathbf{d}_f^T\right]^T$$

Equation 16

[0047] $\tilde{\mathbf{d}}$ is the same as per Equation 8 and $\mathbf{d}_p$ is per Equation 17.

$$\mathbf{d}_p = \left[d(-L+1) \quad d(-L+2) \quad \cdots \quad d(-1)\right]^T \in C^{L-1}$$

Equation 17

[0048] $\mathbf{d}_f$ is per Equation 18.

$$\mathbf{d}_f = [d(N) \quad d(N+1) \quad \cdots \quad d(N+L-2)]^T \in C^{L-1}$$

Equation 18

[0049] The original system model is then per Equation 19 and is illustrated in Figure 4.

$$\mathbf{r} = \mathbf{H}_p\mathbf{d}_p + \tilde{\mathbf{H}}\tilde{\mathbf{d}} + \mathbf{H}_f\mathbf{d}_f + \mathbf{n}$$

Equation 19

[0050] One approach to model Equation 19 is per Equation 20.

$$\tilde{\mathbf{r}} = \tilde{\mathbf{H}}\tilde{\mathbf{d}} + \tilde{\mathbf{n}}_1$$

where $\tilde{\mathbf{r}} = \mathbf{r} - \mathbf{H}_p\mathbf{d}_p$ and $\tilde{\mathbf{n}}_1 = \mathbf{H}_f\mathbf{d}_f + \mathbf{n}$

Equation 20

[0051] Using an MMSE algorithm, the estimated data vector $\hat{\tilde{\mathbf{d}}}$ is per Equation 21.

$$\hat{\tilde{\mathbf{d}}} = g_d\tilde{\mathbf{H}}^H(g_d\tilde{\mathbf{H}}\tilde{\mathbf{H}}^H + \Sigma_1)^{-1}\hat{\tilde{\mathbf{r}}}$$

Equation 21

[0052] In Equation 21, g_d is chip energy per Equation 22.

$$E\{d(i)d^*(j)\} = g_d\delta_{ij}$$

Equation 22

[0053] $\hat{\tilde{\mathbf{r}}}$ is per Equation 23.

$$\hat{\tilde{\mathbf{r}}} = \mathbf{r} - \mathbf{H}_p\hat{\mathbf{d}}_p$$

Equation 23

[0054] $\hat{\mathbf{d}}_p$, is part of the estimation of $\tilde{\mathbf{d}}$ in the previous sliding window step. Σ_1 is the autocorrelation matrix of $\tilde{\mathbf{n}}_1$, i.e., $\Sigma_1 = E\{\tilde{\mathbf{n}}_1\tilde{\mathbf{n}}_1{}^H\}$. If assuming $\mathbf{H}_f\mathbf{d}_f$ and $\mathbf{n}$ are uncorrelated, Equation 24 results.

$$\Sigma_1 = g_d\mathbf{H}_f\mathbf{H}_f^H + E\{\mathbf{n}\mathbf{n}^H\}$$

Equation 24

[0055] The reliability of $\hat{\mathbf{d}}_p$ depends on the sliding window size (relative to the channel delay span L) and sliding step size.

[0056] This approach is also described in conjunction with the flow diagram of Figure 5 and preferred receiver components of Figure 6, which can be implemented in a WTRU or base station. The circuit of Figure 6 can be implemented on a single integrated circuit (IC), such as an application specific integrated circuit (ASIC), on multiple IC's, as discrete components or as a combination of IC('s) and discrete components.

[0057] A channel estimation device 20 processes the received vector $\mathbf{r}$ producing the channel estimate matrix portions, $\mathbf{H}_p$, $\tilde{\mathbf{H}}$ and $\mathbf{H}_f$, (step 50). A future noise auto-correlation device 24 determines a future noise auto-correlation factor, $g_d \mathbf{H}_f \mathbf{H}_f^H$, (step 52). A noise auto-correlation device 22 determines a noise auto-correlation factor, $E\{\mathbf{nn}^H\}$, (step 54). A summer 26 sums the two factors together to produce Σ_1, (step 56).

[0058] A past input correction device 28 takes the past portion of the channel response matrix, $\mathbf{H}_p$, and a past determined portion of the data vector, $\hat{\mathbf{d}}_p$, to produce a past correction factor, $\mathbf{H}_p \hat{\mathbf{d}}_p$, (step 58). A subtractor 30 subtracts the past correction factor from the received vector producing a modified received vector, $\hat{\tilde{\mathbf{r}}}$, (step 60). An MMSE device 34 uses Σ_1, $\tilde{\mathbf{H}}$, and $\hat{\tilde{\mathbf{r}}}$ to determine the received data vector center portion $\hat{\tilde{\mathbf{d}}}$, such as per Equation 21, (step 62). The next window is determined in the same manner using a portion of $\hat{\tilde{\mathbf{d}}}$ as $\hat{\mathbf{d}}_p$ in the next window determination, (step 64). As illustrated in this approach, only data for the portion of interest, $\hat{\tilde{\mathbf{d}}}$, is determined reducing the complexity involved in the data detection and the truncating of unwanted portions of the data vector.

[0059] In another approach to data detection, only the noise term is corrected. In this approach, the system model is per Equation 25.

$$\mathbf{r} = \tilde{\mathbf{H}}\tilde{\mathbf{d}} + \tilde{\mathbf{n}}_2 \text{, where } \tilde{\mathbf{n}}_2 = \mathbf{H}_p\mathbf{d}_p + \mathbf{H}_f\mathbf{d}_f + \mathbf{n}$$

Equation 25

[0060] Using an MMSE algorithm, the estimated data vector $\hat{\tilde{\mathbf{d}}}$ is per Equation 26.

$$\hat{\tilde{\mathbf{d}}} = g_d\tilde{\mathbf{H}}^H(g_d\tilde{\mathbf{H}}\tilde{\mathbf{H}}^H + \Sigma_2)^{-1}\mathbf{r}$$

Equation 26

[0061] Assuming $\mathbf{H}_p\mathbf{d}_p, \mathbf{H}_f\mathbf{d}_f$ and $\mathbf{n}$ are uncorrelated, Equation 27 results.

$$\Sigma_2 = g_d\mathbf{H}_p\mathbf{H}_p^H + g_d\mathbf{H}_f\mathbf{H}_f^H + E\{\mathbf{nn}^H\}$$

Equation 27

[0062] To reduce the complexity in solving Equation 26 using Equation 27, a full matrix multiplication for $\mathbf{H}_p\mathbf{H}_p^H$ and $\mathbf{H}_f\mathbf{H}_f^H$ are not necessary, since only the upper and lower corner of $\mathbf{H}_p$ and $\mathbf{H}_f$, respectively, are non-zero, in general.

[0063] This approach is also described in conjunction with the flow diagram of Figure 7 and preferred receiver components of Figure 8, which can be implemented in a WTRU or base station. The circuit of Figure 8 can be implemented on a single integrated circuit (IC), such as an application specific integrated circuit (ASIC), on multiple IC's, as discrete components or as a combination of IC('s) and discrete components.

[0064] A channel estimation device 36 processes the received vector producing the channel estimate matrix portions, $\mathbf{H}_p, \tilde{\mathbf{H}}$ and $\mathbf{H}_f$, (step 70). A noise auto-correlation correction device 38 determines a noise auto-correlation correction factor, $g_d\mathbf{H}_p\mathbf{H}_p^H + g_d\mathbf{H}_f\mathbf{H}_f^H$, using the future and past portions of the channel response matrix, (step 72). A noise auto correlation device 40 determines a noise auto-correlation factor, $E\{\mathbf{nn}^H\}$, (step 74). A summer 42 adds the noise auto-correlation correction factor to the noise auto-correlation factor to produce Σ_2, (step 76). An MMSE device 44 uses the center portion or the channel response matrix, $\tilde{\mathbf{H}}$, the received vector, $\mathbf{r}$, and Σ_2 to

estimate the center portion of the data vector, $\hat{\tilde{\mathbf{d}}}$, (step 78). One advantage to this approach is that a feedback loop using the detected data is not required. As a result, the different slided window version can be determined in parallel and not sequentially.

*　　　*　　　*